

## Claims

1. A method of mapping virtual memory to physical memory, comprising:
  - identifying two or more contiguous pages in virtual memory to be mapped to physical memory;
  - determining a size in pages of the two or more contiguous pages of virtual memory;
  - determining an alignment in pages of the two or more contiguous pages of virtual memory;
  - searching a free bit data structure to locate a free section of contiguous physical memory having the desired size and alignment; and
  - mapping the two or more pages in virtual memory to the located free section of contiguous memory via a single mapping.
2. The method of claim 1, wherein the size of the two or more contiguous pages of virtual memory and the alignment of the two or more contiguous pages of virtual memory are different values.
3. The method of claim 1, wherein searching a free bit data structure comprises processing a word comprising free bits using a population count instruction to determine the number of free bits in the word.

4. The method of claim 3, wherein the population count instruction is a vector instruction operable to operate on multiple words per instruction issue.
5. The method of claim 1, wherein searching a free bit data structure comprises processing a word comprising free bits using a leading bit count instruction to determine the number of leading bits of a particular value.
6. The method of claim 5, wherein the population count instruction is a vector instruction operable to operate on multiple words per instruction issue.
7. The method of claim 1, wherein searching a free bit data structure comprises searching for a number of consecutive free bits matching the determined size in pages of the two or more virtual memory pages to be mapped.
8. The method of claim 1, wherein searching a free bit data structure comprises counting consecutive free bits, wherein counting the number of free bits found is started only from bits representing a page number that is an integer multiple of the determined alignment.
9. The method of claim 1, wherein the free bit data structure is derived from extraction of free bits from a page status table.

10. A memory management system comprising a virtual memory mapping module, the virtual memory mapping module operable to:

identify two or more contiguous pages in virtual memory to be mapped to physical memory;

determine a size in pages of the two or more contiguous pages of virtual memory;

determine an alignment in pages of the two or more contiguous pages of virtual memory;

search a free bit data structure to locate a free section of contiguous physical memory having the desired size and alignment; and

map the two or more pages in virtual memory to the located free section of contiguous memory via a single mapping.

11. The memory management system of claim 10, wherein the size of the two or more contiguous pages of virtual memory and the alignment of the two or more contiguous pages of virtual memory are different values.

12. The memory management system of claim 10, wherein searching a free bit data structure comprises processing a word comprising free bits using a population count instruction to determine the number of free bits in the word.

13. The memory management system of claim 12, wherein the population count

instruction is a vector instruction operable to operate on multiple words per instruction issue.

14. The memory management system of claim 10, wherein searching a free bit data structure comprises processing a word comprising free bits using a leading bit count instruction to determine the number of leading bits of a particular value.

15. The memory management system of claim 14, wherein the population count instruction is a vector instruction operable to operate on multiple words per instruction issue.

16. The memory management system of claim 10, wherein searching a free bit data structure comprises searching for a number of consecutive free bits matching the determined size in pages of the two or more virtual memory pages to be mapped.

17. The memory management system of claim 10, wherein searching a free bit data structure comprises counting consecutive free bits, wherein counting the number of free bits found is started only from bits representing a page number that is an integer multiple of the determined alignment.

18. The memory management system of claim 10, wherein the free bit data structure is derived from extraction of free bits from a page status table.

19. A computerized system having a paged memory system, the paged memory system operable to:

- identify two or more contiguous pages in virtual memory to be mapped to physical memory;
- determine a size in pages of the two or more contiguous pages of virtual memory;
- determine an alignment in pages of the two or more contiguous pages of virtual memory;
- search a free bit data structure to locate a free section of contiguous physical memory having the desired size and alignment; and
- map the two or more pages in virtual memory to the located free section of contiguous memory via a single mapping.

20. The computerized system of claim 19, wherein the size of the two or more contiguous pages of virtual memory and the alignment of the two or more contiguous pages of virtual memory are different values.

21. The computerized system of claim 19, wherein searching a free bit data structure comprises processing a word comprising free bits using a population count instruction to determine the number of free bits in the word.

22. The computerized system of claim 21, wherein the population count instruction is

a vector instruction operable to operate on multiple words per instruction issue.

23. The computerized system of claim 19, wherein searching a free bit data structure comprises processing a word comprising free bits using a leading bit count instruction to determine the number of leading bits of a particular value.

24. The computerized system of claim 23, wherein the population count instruction is a vector instruction operable to operate on multiple words per instruction issue.

25. The computerized system of claim 19, wherein searching a free bit data structure comprises searching for a number of consecutive free bits matching the determined size in pages of the two or more virtual memory pages to be mapped.

26. The computerized system of claim 19, wherein searching a free bit data structure comprises counting consecutive free bits, wherein counting the number of free bits found is started only from bits representing a page number that is an integer multiple of the determined alignment.

27. The computerized system of claim 19, wherein the free bit data structure is derived from extraction of free bits from a page status table.